```python
import numpy as np
import matplotlib.pyplot as plt

def function(x):
    return (x + 3)**2

def gradient(x):
    return 2 * (x + 3)

def gradient_descent(initial_x, learning_rate, num_iterations):
    x = initial_x
    x_history = [x]
    y_history = [function(x)]

    for i in range(num_iterations):
        grad = gradient(x)
        x = x - learning_rate * grad
        x_history.append(x)
        y_history.append(function(x))

        # Print progress every 10 iterations
        if (i + 1) % 10 == 0:
            print(f"Iteration {i+1}: x = {x:.6f}, y =
{function(x):.6f}, gradient = {grad:.6f}")

    return x, x_history, y_history

initial_x = 2
learning_rate = 0.1
num_iterations = 50

final_x, x_history, y_history = gradient_descent(initial_x,
learning_rate, num_iterations)

print("\n--- Results ---")
print(f"Starting point: x = {initial_x}")
print(f"Final point: x = {final_x:.6f}")
print(f"Local minimum: y = {function(final_x):.6f}")
print(f"Number of iterations: {num_iterations}")
```

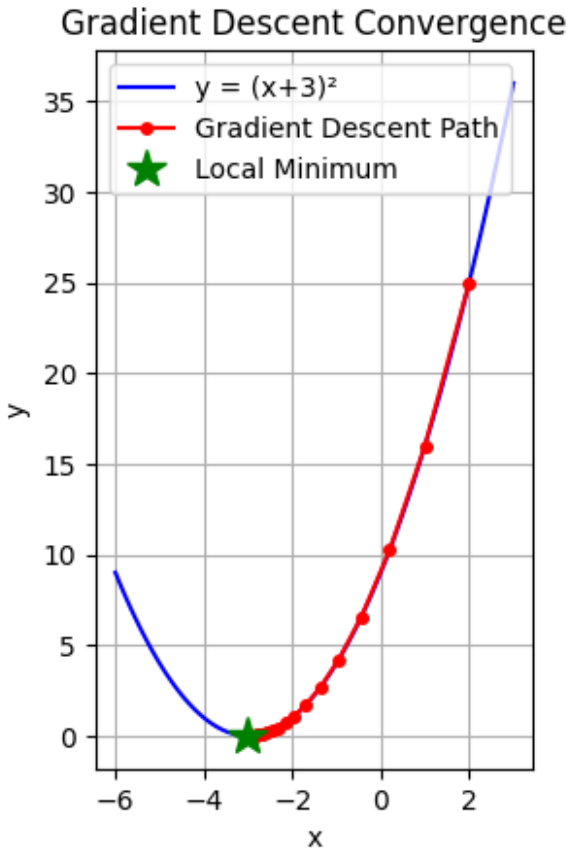
```
Iteration 10: x = -2.463129, y = 0.288230, gradient = 1.342177
Iteration 20: x = -2.942354, y = 0.003323, gradient = 0.144115
Iteration 30: x = -2.993810, y = 0.000038, gradient = 0.015474
Iteration 40: x = -2.999335, y = 0.000000, gradient = 0.001662
Iteration 50: x = -2.999929, y = 0.000000, gradient = 0.000178

--- Results ---
Starting point: x = 2
Final point: x = -2.999929
Local minimum: y = 0.000000
Number of iterations: 50
```

```
plt.figure(figsize=(12, 5))
```

```
<Figure size 1200x500 with 0 Axes>
```

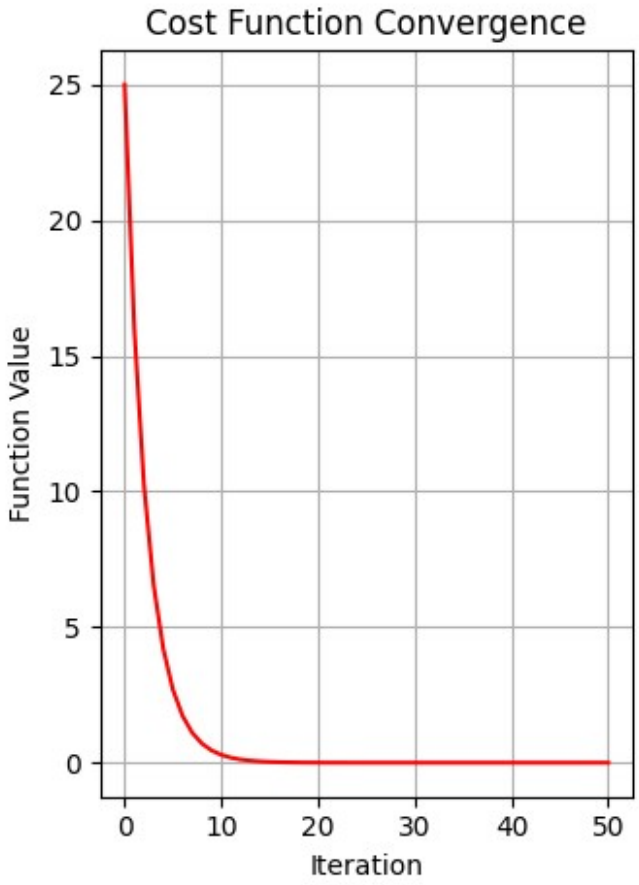
```
<Figure size 1200x500 with 0 Axes>
```

```
plt.subplot(1, 2, 1)
x_range = np.linspace(-6, 3, 100)
y_range = function(x_range)
plt.plot(x_range, y_range, 'b-', label='y = (x+3)²')
plt.plot(x_history, y_history, 'ro-', markersize=4, label='Gradient
Descent Path')
plt.plot(final_x, function(final_x), 'g*', markersize=15, label='Local
Minimum')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Gradient Descent Convergence')
plt.legend()
plt.grid(True)
```




```
plt.subplot(1, 2, 2)
plt.plot(range(len(y_history)), y_history, 'r-')
plt.xlabel('Iteration')
```

```
plt.ylabel('Function Value')
plt.title('Cost Function Convergence')
plt.grid(True)

plt.tight_layout()
plt.show()
```



Iteration 10: x = -2.463129, y = 0.288230, gradient = 1.342177 Iteration 20: x = -2.942354, y = 0.003323, gradient = 0.144115 Iteration 30: x = -2.993810, y = 0.000038, gradient = 0.015474 Iteration 40: x = -2.999335, y = 0.000000, gradient = 0.001662 Iteration 50: x = -2.999929, y = 0.000000, **gradient** = 0.000178

--- Results --- Starting point: x = 2 Final point: x = -2.999929 Local minimum: y = 0.000000 Number of iterations: 50
